Tennessee Tech UNIVERSITY

Time-Reversal Based Range Extension Technique for Ultra-wideband (UWB) Sensors and Applications in Tactical Communications and Networking

Technical Report (Quarterly)

to

US Office of Naval Research

875 North Randolph Street

Arlington, VA 22203-1995

for

Grant # N00014-07-1-0529

Prepared by

Robert C. Qiu

(Principal Investigator)

together with

(Contributing Researchers at Wireless Networking Systems Lab)

Nan (Terry) Guo
Yu Song
Peng (Peter) Zhang
Zhen (Edward) Hu

January 16, 2009

Department of Electrical and Computer Engineering

Center for Manufacturing Research

Tennessee Technological University

Cookeville, TN 38505

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
16-01-2009	Technical Report (Quarterly)	July 16, 2008---January 16, 2009

4. TITLE AND SUBTITLE

Time-Reversal Based Range Extension Technique for Ultra-wideband (UWB) Sensors and Applications in Tactical Communications and Networking

5a. CONTRACT NUMBER

5b. GRANT NUMBER

N00014-07-1-0529

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Qiu, Robert C.; Guo, Nan; Hu, Zhen; Zhang, Peng; Yu, Song; Chen, Zhe

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Tennessee Technological University
115 W. 10th Street
Cookeville, TN 38501

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

US Office of Naval Research
875 North Randolph Street
Arlington, VA 22203-1995

10. SPONSOR/MONITOR'S ACRONYM(S)

ONR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

07PR05074-00

12. DISTRIBUTION/AVAILABILITY STATEMENT

release for public distribution.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report summarizes our research and development work in the last quarter of year 2008. We have reached a practical limit in terms of sampling rate and bandwidth. Listed below are major achievements during the quarter:

- improved waveform generator: FPGA coding with respect to readability, quantization at the transmitter has been increased from ternary to 8-bit, and sampling rate has been increased from 500 Ms/s to 1 Gs/s;
- I/Q quadrature modulation/upconversion, instead of single-channel;
- Xilinx Virtex 5 LXT board at the receiver with new high speed connection to the A/D;
- variable thresholds for various signal levels;
- 10-dB Bandwidth up to 800 MHz;
- multiple bit rates: 6.25, 3.125, and 1.5625 Mb/s;
- improved receiver sensitivity.

15. SUBJECT TERMS

UWB, testbed, time reversal, range extension, sensors, multi-GHz wideband, cognitive radio

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Francis Otuonye
U	U	U	UU	24	19b. TELEPHONE NUMBER *(Include area code)* 931-372-3374

Acknowledgment

This work has been improved by discussions with S. K. Das (ONR), B. M. Sadler (ARL), R. Ulman (ARO), and L. Lunardi (formerly with NSF). K. Currie (CMR, TTU) has provided a lot of support for this project. S. Parke (ECE, TTU) has supported our research in different ways. We also want to thank P. K. Rajan for helpful discussions.

Executive Summary

This report summarizes our research and development work in the last quarter of year 2008. We have reached a practical limit in terms of sampling rate and bandwidth. Listed below are major achievements during the quarter:

- improved waveform generator: FPGA coding with respect to readability, quantization at the transmitter has been increased from ternary to 8-bit, and sampling rate has been increased from 500 Ms/s to 1 Gs/s;
- I/Q quadrature modulation/upconversion, instead of single-channel;
- Xilinx Virtex 5 LXT board at the receiver with new high speed connection to the A/D;
- variable thresholds for various signal levels;
- 10-dB Bandwidth up to 800 MHz;
- multiple bit rates: 6.25, 3.125, and 1.5625 Mb/s;
- improved receiver sensitivity.

Our next milestone, a gain of several dB over ordinary non-time-reversal system, is anticipated, and system trials will be reported in the April report.

Contents

Part I

Testbed Development

Chapter 1

Unique System Design Issues

1.1 Passband Waveform Precoding

A flexible way to generate passband signal is to use a dual-channel baseband waveform generator followed by an I/Q up-converter (modulator). In this way, the spectral shape can be easily controlled by changing the local oscillator's frequency and the baseband waveforms. Considering the nature of the energy detector receiver, the goal of transmit waveform precoding is to achieve a desired signal envelope at the input to the detector. If $w(t)$ and $h(t)$ are the baseband representations of the transmit chip waveform and the channel impulse response (CIR), respectively, then the transmit waveform and the CIR can be expressed as

$$\tilde{w}(t) = Re\{w(t)e^{-j2\pi f_c t}\}, \tag{1.1}$$
$$\tilde{h}(t) = Re\{h(t)e^{-j2\pi f_c t}\}, \tag{1.2}$$

where f_c is the center frequency. It can be proved that the received chip waveform is given by

$$\begin{aligned} \tilde{y}(t) &= \tilde{w}(t) \otimes \tilde{h}(t) \\ &= Re\{[\frac{1}{2}w(t) \otimes h(t)]e^{-j2\pi f_c t}\} \\ &= Re\{y(t)e^{-j2\pi f_c t}\}, \end{aligned} \tag{1.3}$$

where $\otimes$ is convolution operator and $y(t) = \frac{1}{2}w(t) \otimes h(t)$ is the baseband representation of the received chip waveform without noise pollution. In other words, the transmitter-receiver chain can be represented equivalently in baseband as illustrated in Fig.1.1, where the factor of 1/2 (see above equation)has been absorbed into $h(t)$. Note that $h(t)$ takes into account the impacts of propagation channel and RF front-ends including antennas at both sides.

1.2 Estimation of the Equivalent Filter in the Transmitter-Receiver Chain

The equivalent baseband filter in the transmitter-receiver chain has to be taken into account in order to have optimum waveform at the receiver detector. This filter is generally not flat over the frequency band of interest, especially for wide bandwidth case. Estimating and controlling the equivalent baseband filter is not easy. As shown in Fig.1.2, the

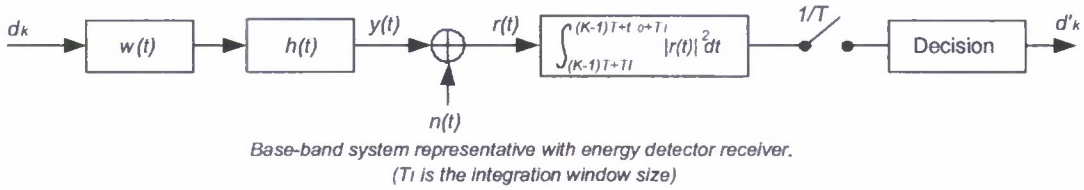

Figure 1.1: Baseband system equivalence.

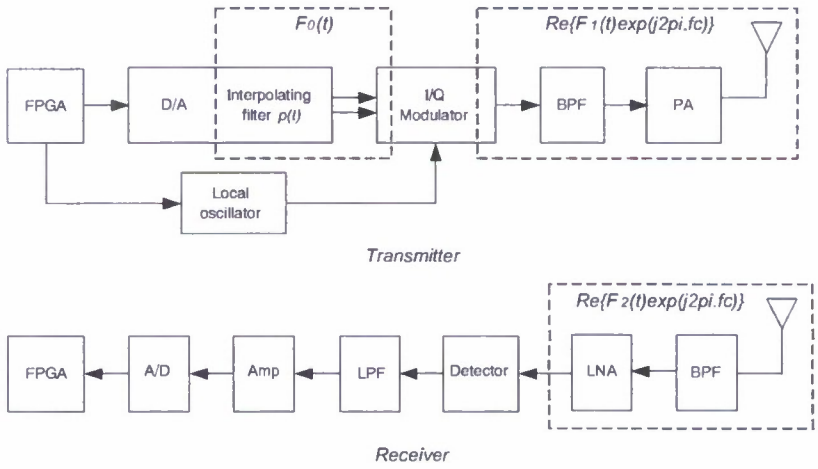

Figure 1.2: Equivalent filter in transmitter-receiver chain (not including the propagation channel), contributed from three sources.

overall filter effect is contributed from three sources: the D/A's interpolating filter $p(t)$ and RF front-ends including antennas at both sides. By omitting some scalar factor, the equivalent filter in baseband format is

$$F(t) = F_0(t) \otimes F_1(t) \otimes F_2(t), \tag{1.4}$$

and $F_0(t)$ can be expressed using the interpolating filter $p(t)$

$$F_0(t) = p(t) + j \cdot p(t). \tag{1.5}$$

According to our measurement, the shape of the equivalent filter is dominated by the interpolating filter $p(t)$. Based on the A/D used in the testbed, $p(t)$ has been accurately measured and $F_0(t) = p(t) + j \cdot p(t)$ has been used as an estimate of $F(t)$.

1.3 Discrete-Time Representation

Practically speaking, precoding is performed in the digital back-end and the baseband waveform should be represented in digital format (discrete-time and quantized) by x_m at sampling rate $1/T_S$. Similarly, $h(t)$ can be represented in discrete-time by h_n at sampling rate M_1/T_S, $M_1 \geq 1$. It is assumed that the sampling rate $1/T_S$ is determined by the hardware limitation, and the sampling rate M_1/T_S is no less than the Nyquist rate based on $h(t)$.

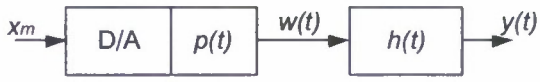

Figure 1.3: Continuous-time waveforms generated from discrete-time waveforms (sequences).

Fig.1.3 is a simplified diagram showing the continuous-time transmit waveform $w(t)$ and receive waveform $y(t)$ originate from the digital waveform x_m. It is necessary to find how $y(t)$ is calculated from sequences x_m and h_n.

Let us first generate an analytical up-sampling signal $x'_{m'}$ corresponding to x_m:

$$x'_{m'} = \begin{cases} x_m, & \text{if } m' = M_1 m \\ 0, & \text{otherwise} \end{cases} \tag{1.6}$$

so that $x'_{m'}$ and h_n have the same sampling rate M_1/T_S. By using the ideal interpolating function $sinc(M_1 t/T_S)$, the target waveform $y(t)$ can be expressed as

$$\begin{aligned} y(t) &= \left[\sum_{m'} x'_{m'} p\left(t - \frac{m'T_S}{M_1}\right)\right] \otimes \left[\sum_{n} h_n sinc\left(\frac{M_1 t}{T_S} - n\right)\right] \\ &= \sum_{m'}\sum_{n} x'_{m'} h_n \int_{-\infty}^{\infty} p\left(\tau - \frac{m'T_S}{M_1}\right) sinc\left(\frac{M_1(t-\tau)}{T_S} - n\right) d\tau \end{aligned} \tag{1.7}$$

Let $\nu = \tau - m'T_S/M_1$ and define

$$q(t) = p(t) \otimes sinc(M_1 t/T_S), \tag{1.8}$$

then $y(t)$ can be further expressed as

$$\begin{aligned} y(t) &= \sum_{m'}\sum_{n} x'_{m'} h_n \int_{-\infty}^{\infty} p(\nu) sinc\left(\frac{M_1(t-\nu)}{T_S} - (m'+n)\right) d\nu \\ &= \sum_{m'}\sum_{n} x'_{m'} h_n q\left(t - \frac{(m'+n)T_S}{M_1}\right) \\ &= \sum_{k}\sum_{m'} x'_{m'} h_{k-m'} q\left(t - \frac{kT_S}{M_1}\right) \\ &= \sum_{k}\sum_{m} x_m h_{k-mM_1} q\left(t - \frac{kT_S}{M_1}\right), \end{aligned} \tag{1.9}$$

where $k = m' + n$ has been applied. It is worth noting that in the above equation both $x'_{m'}$ and h_n are in complex format, $\sum_m x_m h_{k-mM_1}$ represents discrete-time convolution, and $q(t)$ is a non-ideal interpolating function.

The target waveform can also be represented based on the equivalent CIR $\hat{h}(t)$ defined as

$$\begin{aligned} \hat{h}(t) &= p(t) \otimes h(t) \\ &= \sum_{n} \hat{h}_n sinc\left(\frac{M_2 t}{T_S} - n\right), \end{aligned}$$

where $\hat{h}_n$ is a discrete-time version of $\hat{h}(t)$, sampled at the Nyquist rate (or above that) M_2/T_S. Define $x''_{m''}$ as

$$x''_{m''} = \begin{cases} x_m, & \text{if } m'' = M_2 m \\ 0, & \text{otherwise} \end{cases} \tag{1.10}$$

then $y(t)$ can be expressed in a similar form:

$$\begin{aligned} y(t) &= \sum_k \sum_{m''} x''_{m''} \hat{h}_{k-m''} sinc\left(\frac{M_2 t}{T_S} - k\right) \\ &= \sum_k \sum_m x_m \hat{h}_{k-mM_2} sinc\left(\frac{M_2 t}{T_S} - k\right). \end{aligned} \tag{1.11}$$

1.4 Balanced I/Q Quantization

x_m is actually a digitized complex signal with certain quantization resolution. To make full use of the A/D's quantization capability, it is desired to balance the peak values in the real and image parts, and use the peak value as the D/A's full scale. When energy detector receiver is used, the waveform precoding is to achieve a desired envelope $y_{opt}(t)$ at the receiver detector. If this balance requirement is not met, we can always equalize the two peaks by applying phase rotation to the original waveform $x_{0,m}$ which is in complex format:

$$x_{0,m} = x^I_{0,m} + j \cdot x^Q_{0,m}. \tag{1.12}$$

If there is no waveform overlapping between successive chips, after phase rotation by an angle of θ, the new waveform is

$$\begin{aligned} x_m &= x_{0,m} e^{j\theta} \\ &= x^I_{0,m} cos\theta + j \cdot x^Q_{0,m} sin\theta \\ &= x^I_m + j \cdot x^Q_m. \end{aligned} \tag{1.13}$$

By solving the following equation for θ:

$$max\{|x^I_{0,m}|\} cos\theta = max\{|x^Q_{0,m}|\} sin\theta, \tag{1.14}$$

θ is given by

$$\theta = tan^{-1}\left(\frac{max\{|x^Q_{0,m}|\}}{max\{|x^I_{0,m}|\}}\right). \tag{1.15}$$

When waveform overlapping between successive chips is inevitable, assume the $X_{0,m}$ is a composition of successive chip waveforms before phase rotation:

$$X_{0,m} = X^I_{0,m} + j \cdot X^Q_{0,m}, \tag{1.16}$$

then the rotation angle is

$$\theta = tan^{-1}\left(\frac{max\{|X^Q_{0,m}|\}}{max\{|X^I_{0,m}|\}}\right). \tag{1.17}$$

Chapter 2

Implementation Aspect

2.1 Arbitrary Waveform Generator

Digital Arbitrary Waveform Generator (AWG) is the main part in the backend at the transmitter. Digital AWG can generate virtually any type of transmitted waveform according to our objective, which greatly increases the capability of UWB system. Compared with our first generation real-time UWB testbed, Digital AWG in the current generation real-time UWB testbed is significantly improved. Specifically speaking, the quantization resolution is increased from 2 bits to 8 bits. The sampling rate of output digital data is increased from 500MHz to 1 GHz. The number of channels is increased from 1 to 2. Thus I/Q data can be generated simultaneously. The number of waveform coefficients is increased from 40 to 160. Data memory is added into Digital AWG. Data memory plus 8 groups of shift registers are used to store the waveform coefficients. In this way, the loading of waveform coefficients into FPGA is much easier than before. Look-up table is replaced by Boolean operation and mathematical computation which can support the calculation of input data with higher quantization resolution.

How can we use FPGA with 550MHz maximum clock rate to generate the output digital data with 1GHz sampling rate? Parallel to Serial Converter gives us a hope. The overall structure of digital AWG for one channel is shown in Figure 2.1. Waveform coefficients are stored in Data Memory. When FPGA powers on, waveform coefficients can be automatically loaded from Data Memory to the corresponding group of shift registers. There are 8 groups of shift registers. And each group of shift registers corresponds to one Process Module. The clock rate of each group of shift registers plus the corresponding Process Module is 125 MHz. Finally Parallel to Serial Converter is used to generate the output digital data with 1GHz sampling rate.

For each channel, we have 8 groups of shift registers and each group can store 20 waveform coefficients shown in Figure 2.2. So totally we can have 160 waveform coefficients for each channel.

In the first generation real-time UWB testbed, we used look up table in Process Module. Now we use the multiplication and addition shown in Figure 2.3. In this way, the Verilog code is more compact and readable. The calculation of input data with higher quantization resolution can be supported.

Format Regulation uses Boolean operation to generate complementary code for the following processing in Process Module according to the scrambling code generated by Scrambling Code Generator and the value of input chip.

The key module in digital AWG is the high speed Parallel to Serial Converter, the structure of which is shown in Figure 2.4. Through this way, the high speed requirement of the output digital data is relaxed by parallel computation and Parallel to Serial Converter. The function of parallel to serial conversion is implemented by OSERDES logical resources in Virtex-5 FPGA. Each OSERDES logical component can support 6 to 1 parallel to serial conversion, but we need 8 to 1 parallel to serial conversion, so two components are used to build one converter. One component is called master and one other is slave. The function diagram is shown in Figure 2.5. In our design, the sampling rate of input data from 8 parallel branches is 125 MHz and sampling rate of the output digital data is 1 GHz.

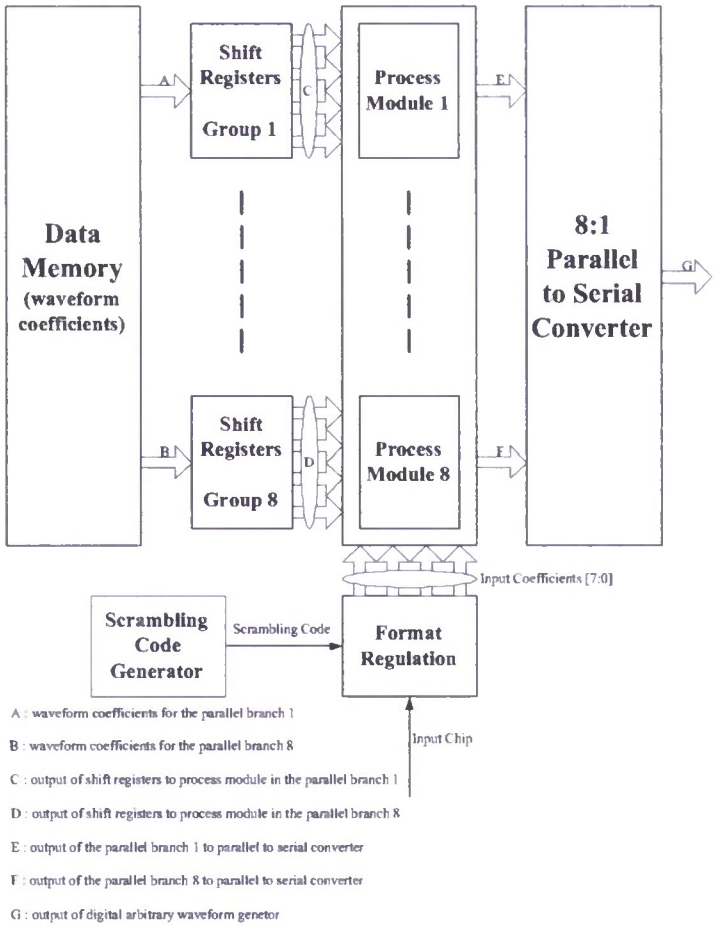

Figure 2.1: Overall structure of digital AWG for one channel.

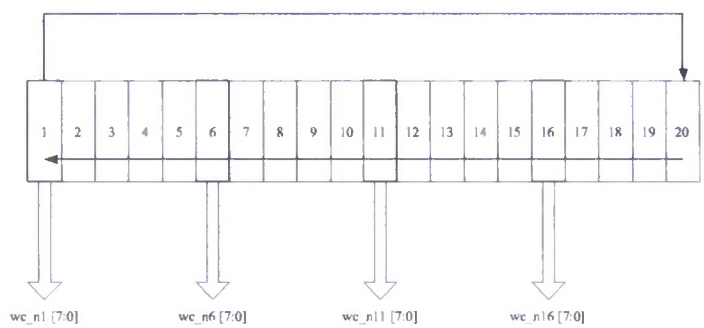

Figure 2.2: Digram of shift registers.

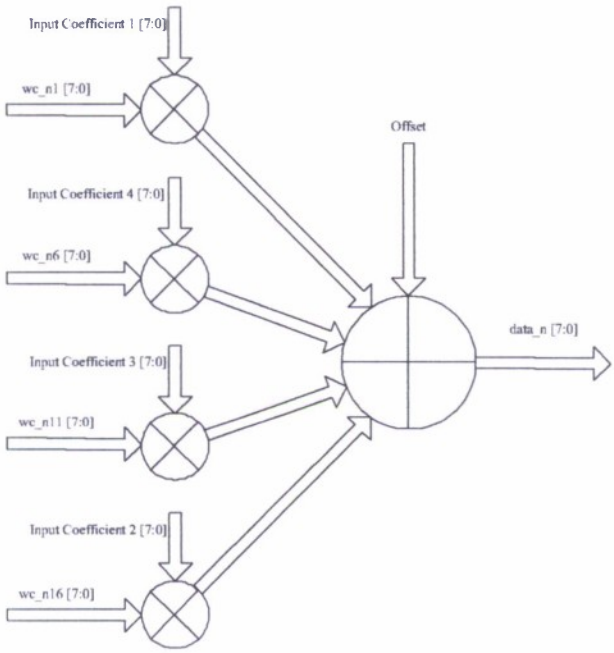

Figure 2.3: Strucuture of Process Module.

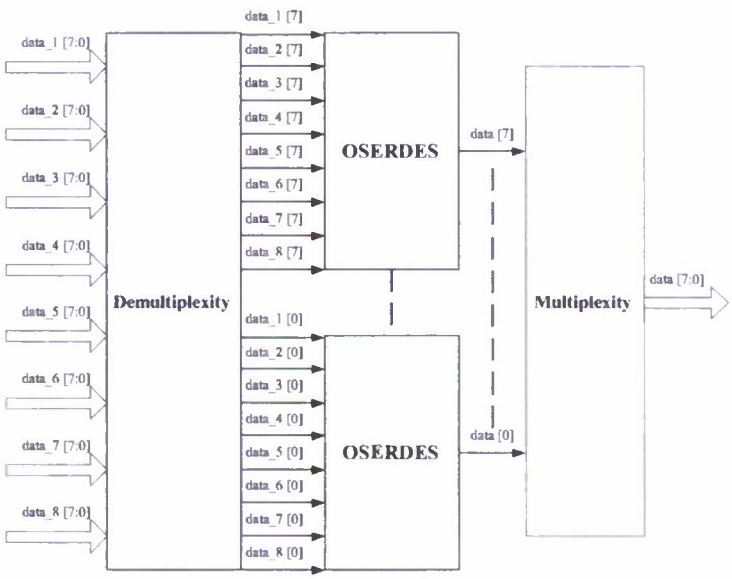

Figure 2.4: Strucuture of Parallel to Serial Converter.

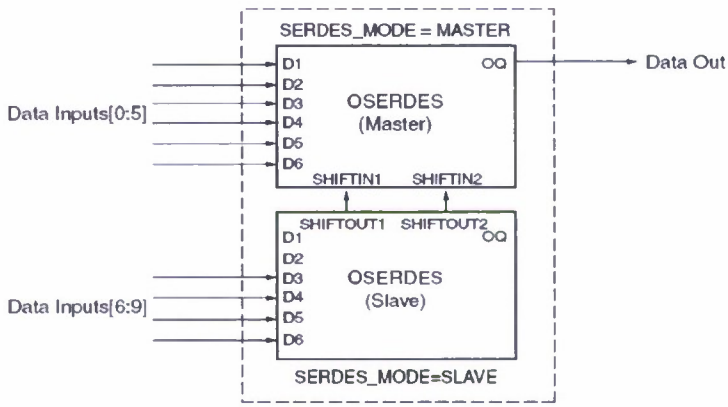

Figure 2.5: Strucuture of OSERDES expansion.

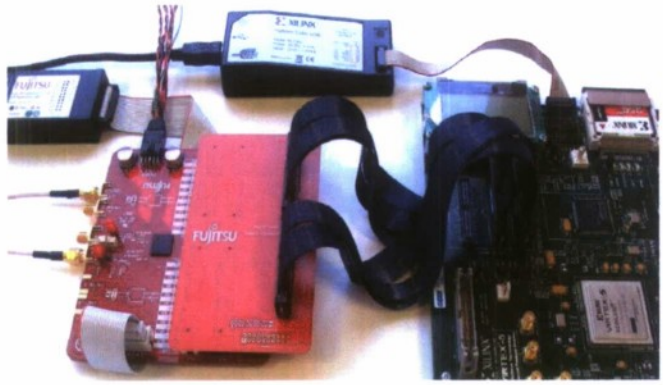

Figure 2.6: Interfacing DAC evaluation board and FPGA evaluation board.

2.2 DAC

The high speed interface between the DAC and FPGA is a critical issue in the testbed. The interface quality of our last testbed could only meet single channel, 500 Msps and 2-bit quantization. This configuration is far below the maximum performance provided by Fujitsu DAC, featuring dual channel, 1 Gsps and 14-bit quantization. In order to fully use the Fujitsu DAC, we found a interfacing solution, with the Xilinx ML550 FPGA evaluation board, an interface adapter built by Fujitsu and a blue ribbon cable (Fig. 2.6).

ML550 provides 6 banks of LVDS I/O (a total of 53 differential input and 53 differential output), supporting up to 1.2 Gbps high data rate transmission. All the I/Os are synchronized. The maximum data rate is limited by the LX50T FPGA, which is just enough to drive the 1 Gsps DAC. If we want to drive a even faster DAC, to the best of our knowledge, parallel to serial conversion is built inside the DAC chip, resulting a reduced burden at the data driving side. For example, the MAXIM 4.3 Gsps DAC uses a 4:1 MUX to drive the DAC, and therefore each of the 4 data channels requires a 1.075 Gsps sampling rate only. As a result, the ML550 I/O interface is capable of driving the highest sampling rate DAC. The blue ribbon cable is a high speed cable provided by Samtec. It supports up to 2.84 Gbps data rate, which is far beyond the our maximum data rate requirement. The interface adapter is customized by Fujitsu. It connects the Fujitsu DAC and the ribbon cable. The pin assignments, a.k.a. mappings between the Fujitsu DAC input ports and FPGA output ports are tested and built as a user constraint file. It can be loaded in the FPGA design and drive the DAC properly.

The purpose of the interface is to fully utilize the performance of the DAC, which is the dual channel, 1 Gsps and 14-bit quantization. A test environment is built to test the performance. All DAC outputs are driven from FPGA with the interface solution. FPGA driven waveforms are compared with the waveform memory module driven ones, since the latter ones are considered to be correct. Fig. ?? shows the two FPGA driven ramp waveforms generated by DAC ports A and B. We can see that two ports are synchronized and the waveforms are the same as the ones generated by waveform memory. A 14-bit mono-cycle 1 ns pulse is generated, as shown in Fig. 2.7. The pulse width is over 1 ns and there is a tail behind the pulse. This might be caused by the circuit of the DAC evaluation board. The spectrum of the mono-cycle pulse is depicted in Fig. 2.8. Another result is that the lower 9-14 bits of the DAC are under the noise level and can not be observed. So, the configuration of the DAC becomes dual channel, 1 Gsps

and 8-bit quantization, which is a great improvement to the previous one.

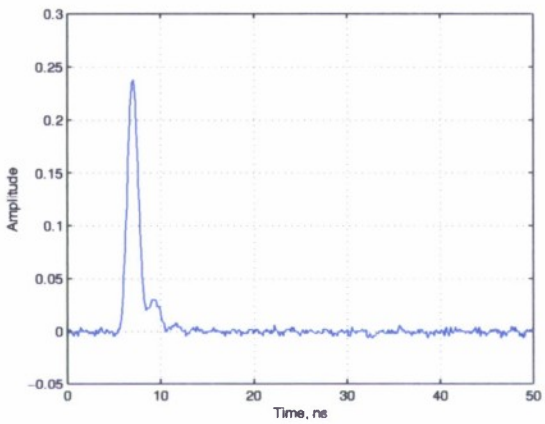

Figure 2.7: A 1 ns pulse generated by DAC.

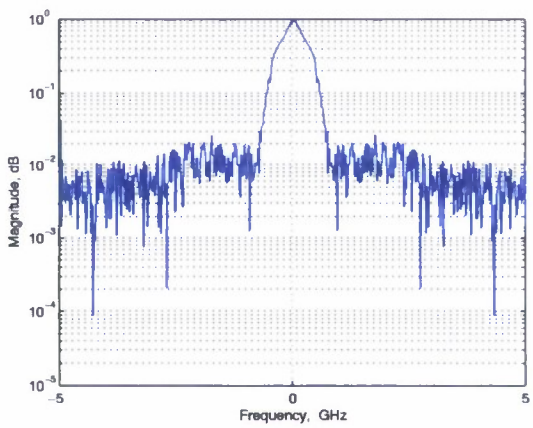

Figure 2.8: The spectrum of a 1 ns pulse generated by DAC.

Chapter 3

Testbed Configuration Issues

For our test-bed development, we have a lot of FPGA and DAC configuration files for different working cases, the following part serves as a memorandum for the test-bed's configurations.

3.1 FPGA configuration files at transmitter side

The test-bed's tansmitter configuration files mainly include "ML550_SPI_LO" and "Tir_ML550" :

- ML550_SPI_LO.

Descriptions: This is the SPI protocol control project files folder at transmitter side.

Functions:

(1),it generates a serial data streams output to configure the PSA4000A local oscillator to work at exactly 4.0 Ghz.

(2), Provide 500MHz output clock for DAC board, this is useful when we only use DAC memory to transmit time reversal waveform to observe the focused peak at receiver.

Sources files list:

(1), spi_lo.v, it serves as the main module.

(2), pll.v, it is used to provide proper working clock signals.

(3), spi_lo.ucf, which defines the input and output pins assignment.

Pins input and output:

NET "le" LOC = "AK32"; connected to P9-8 on FPGA board
NET "mosi" LOC = "AH32"; connected to P9-6 on FPGA board

NET "sclk" LOC = "AK33"; connected to P9-4 on FPGA board
NET "led_show" LOC = "AJ7";
NET "waveform_clk_500_p" LOC = "AD10";
NET "waveform_clk_500_n" LOC = "AD11";

key parameters:

(1), The main working clock is 4MHz, which works for SPI control signals generation.

(2), The output clock for DAC board is 500MHz, where a differential buffer "'OBUFDS"' is used.

(3), The 4 SPI configuration registers value can be changed to configure LO works at other frequencies form 3975MHz to 4025MHz, where register "'N_Latch"' is the critical one, the related information can be found in ADF4106 frequency synthesizer data sheet.

Notes: Each time the local oscillator is powered off, it should be configured again when power on next time.

- Tir_ML550.

Descriptions: This is the main project files folder in transmitter.

Functions:

(1),Output transmitted digital data frame by frame, the frame structure can be found in references.

(2) it provides 500MHz clock signal for DAC when FPGA are used for transmitting.

(3), it generates synchronization signals for digital oscilloscope or other equipments in receiver side.

Sources files list:
(1), OOK_Tx_ML550_V8.v. this is the top module at transmitter side.

(2), sampler.v. this module is used to store transmitted data in RAM and output the frame to encoder module.

(3) encoder.v. this is the function module for channel encoding, it is empty currently.

(4), modulator_1.v. it is a function block for modulation consisting of pulse repetition and OOK modulation.

(5), ss.v. this is the block for spread spectrum, it's empty currently.

(6), pulser.v. it is used to generate short pulses or chips at certain chip rate.

(7), ram_load_v2.v, this is the module for storing waveform source data, and to be feed to waveform generator module.

(8), waveform_generator_v2.v, this is the function module used to generate waveforms need based on transmitted chips and stored waveform sources.

(9), oserdes81_8bit.v, this is the 8 to 1 parallel to serial conversion module, and the data width is 8 bits. (9), oserdes81.v, this is the submodule for oserdes81_8bit.v, the conversion is 1 bit.

(10), pll.v, it the clock management module used to provide various clock signals for other modules.

(11), OOK_Tx_ML550_V8.ucf, this is the top constraint file at transmitter side, it defines the input and output pins assignment and apply various timing constraints for the project.

pins input and output:

```
NET "board_clk_p" LOC = "H17";
NET "board_clk_n" LOC = "H18";
NET "rst_sw" LOC = "W34";
NET "waveform_clk_500_p" LOC = "AD10";
NET "waveform_clk_500_n" LOC = "AD11";
NET "system_clk_125_sync" LOC = "J10";
NET "system_clk_125_sync" IOSTANDARD = "LVCMOS25";
NET "waveform_data_I_p[0]" LOC = "AG28";
NET "waveform_data_I_n[0]" LOC = "AH28";
NET "waveform_data_I_p[1]" LOC = "AD24";
NET "waveform_data_I_n[1]" LOC = "AE24";
NET "waveform_data_I_p[2]" LOC = "AE27";
NET "waveform_data_I_n[2]" LOC = "AE26";
NET "waveform_data_I_p[3]" LOC = "AC25";
NET "waveform_data_I_n[3]" LOC = "AC24";
NET "waveform_data_I_p[4]" LOC = "AB27";
NET "waveform_data_I_n[4]" LOC = "AC27";
NET "waveform_data_I_p[5]" LOC = "AB25";
NET "waveform_data_I_n[5]" LOC = "AB26";
NET "waveform_data_I_p[6]" LOC = "AB28";
NET "waveform_data_I_n[6]" LOC = "AA28";
NET "waveform_data_I_p[7]" LOC = "Y24";
NET "waveform_data_I_n[7]" LOC = "AA24";
NET "waveform_data_Q_p[0]" LOC = "AD30";
NET "waveform_data_Q_n[0]" LOC = "AC29";
NET "waveform_data_Q_p[1]" LOC = "AF29";
NET "waveform_data_Q_n[1]" LOC = "AF30";
NET "waveform_data_Q_p[2]" LOC = "Y26";
NET "waveform_data_Q_n[2]" LOC = "W26";
NET "waveform_data_Q_p[3]" LOC = "Y28";
NET "waveform_data_Q_n[3]" LOC = "Y29";
NET "waveform_data_Q_p[4]" LOC = "V25";
NET "waveform_data_Q_n[4]" LOC = "W25";
NET "waveform_data_Q_p[5]" LOC = "V30";
NET "waveform_data_Q_n[5]" LOC = "W30";
NET "waveform_data_Q_p[6]" LOC = "W24";
NET "waveform_data_Q_n[6]" LOC = "V24";
NET "waveform_data_Q_p[7]" LOC = "Y27";
```

NET "waveform_data_Q_n[7]" LOC = "W27";

Key parameters:

(1), The key parameters in the transmitter is the waveform sources in module "'ram_load_v2.v"', it is loaded in the RAMs, there are two RAMs in this module, one is used to store I channel data, and the other Q channel data. the data length is currently up to 160ns, data width is 8 bits. the data format can be shown as following, where there are 5 lines , each line 32 points. we can either transmit 160 points time reversal waveforms or just transmit 2ns pulse. The data order starts form the right 8 bits of the first line("'09"' in the following example"'), after ends in the left side, it starts from the right side the second line("'09"' in the following example"') and likewise. These data are automatically read out and feed to other modules when the project is run.

.INIT_00(256'h141404FA02EDE20705FDEEF40813FCF408000B0FF8F7F5F6FC0C04E7EBFBF009),
.INIT_01(256'hD4E415FAFF03E6CAD5B0E629F20218FE12151E320A1D2AFE241D12131C20F911),
.INIT_02(256'h0000000000000000000000000000000010214D37530AF942CFF219EE9CABC1C2),
.INIT_03(256'h0000000000000000000000000000000000000000000000000000000000000000),
.INIT_04(256'h0000000000000000000000000000000000000000000000000000000000000000),

(2), In modulator_1.v module, we can change the 60 bits frame synchronization codes between OOC or M sequence.

(3), In waveform_generator_v2.v module, we send 200ns dense pulses for start pattern searching before each frame, and this part can be modified. The format is controlled by the control signals "'tt_w"' and "'t_w"',where the first one represents the number of symbols(1024) for each frame and the later one represents the number of 40 ns of each chip.

(4), The DAC sampling rate is controlled by the clock signal 'waveform_clk_500" which is 500MHz, if we change this clock rate, other clocks rate should be changed accordingly, the recommendation is to change the overall parameters in pll.v module by modifying the multiplication factor for all output clocks.

(5), Change between using 4 chips or 16 chips to represent one bit. this can be changed in modulator_1.v module.

Notes: more details can be found in file "'log.doc"'.

3.2 FPGA configuration files at receiver side

- OOK_Rx_Synch_V5_TiR.

Descriptions: This is the main project files folder in receiver.

Functions:

(1),it receives digital data streams from ADC board at high speed of 800 Mb/s, the data width is 8 bits with both primary channel and auxiliary channel.

(2), Synchronizes the received frame.

(3), Demodulates the data bits sent in the transmitter.

Sources files list:

(1), OOK_Rx_Synch_V80_top.v. This is the top module in receiver.

(2), Interface_1.v. this is the interface module which receive data streams from ADC, and reduce the high speed data to relatively low speed data streams through techniques such as FIFO, DDR, serial to parallel conversion. submodules includes HSRX_16D_2TO1.v, DDR_REG.v, fifo_64to64.v, DDR_REG.v, BUF_LUT.v and pllsecond.v.

(3), integration_7.v. this is the integration module, it implements chip level energy integration by using 32 parallel DSP48E based integrators, the results are stored in 32 registers with each corresponds to an integrator. it includes two submodules accumulator_2.v, accu_v7_ip_500_ise9.v.

(4), start_pattern.v. this moudle is used to search the arrival of the frame by comparing the integration results with certain threshold.

(5), synch_searching_1.v. This is the synchronization module, which combines the time of arrival signal from start pattern searching and the integration results to get a chip threshold by averaging 128 chips, then it synchronize the 60 bits M sequence data, after that the whole frame is synchronized. it includes a submodule Maximun_6.v.

(6), data_process_1.v. this module accumulates 4 chips integration result from integration module, the result means each received data value since we represent each bit by 4 chips.

(7), detection.v. this is the detection module which compares the threshold achieved in synchronization module and the data value from data process module, and the result is the demodulation result, which should be the data bits sent in transmitter.

(8), FSM.v, this is the finite state machine module, which coordinate the implementation order of all other process.

(9), Variable_threshold4.v, this is the variable threshold module, which can change the threshold value by push buttons on FPGA board. in includes 4 thresholds currently, the start pattern threshold, the chip threshold ratio, the frame level synchronization threshold and the ADC input bias shift.

(10), pllmain.v. this is the clock management module which provides various clock signals for all modules.

Pins input and output:

```
NET "board_clk" LOC = E18;
NET "data_out" LOC = E13;
NET "rst_sw" LOC = AD14;
NET "REFCLKN" LOC = D8;
NET "REFCLKP" LOC = C9;
NET "synch_flag_1" LOC = H26;
NET "RXN_data_A_0" LOC = F8;
NET "RXP_data_A_0" LOC = F7;
NET "RXN_data_A_1" LOC = G9;
NET "RXP_data_A_1" LOC = F9;
NET "RXN_data_A_2" LOC = J8;
```

```
NET "RXP_data_A_2" LOC = H8;
NET "RXN_data_A_3" LOC = A8;
NET "RXP_data_A_3" LOC = A9;
NET "RXP_data_A_4" LOC = E8;
NET "RXN_data_A_4" LOC = E7;
NET "RXN_data_A_5" LOC = C8;
NET "RXP_data_A_5" LOC = B9;
NET "RXP_data_A_6" LOC = E6;
NET "RXN_data_A_6" LOC = D6;
NET "RXN_data_A_7" LOC = G7;
NET "RXP_data_A_7" LOC = H7;
NET "RXN_data_P_0" LOC = C6;
NET "RXP_data_P_0" LOC = C7;
NET "RXP_data_P_1" LOC = A7;
NET "RXN_data_P_1" LOC = B7;
NET "RXN_data_P_2" LOC = D10;
NET "RXP_data_P_2" LOC = D9;
NET "RXP_data_P_3" LOC = B6;
NET "RXN_data_P_3" LOC = A5;
NET "RXN_data_P_4" LOC = A10;
NET "RXP_data_P_4" LOC = B10;
NET "RXP_data_P_5" LOC = A4;
NET "RXN_data_P_5" LOC = A3;
NET "RXN_data_P_6" LOC = A12;
NET "RXP_data_P_6" LOC = B11;
NET "RXP_data_P_7" LOC = B4;
NET "RXN_data_P_7" LOC = B5;
NET "state_0" LOC = F24;
NET "state_1" LOC = G25;
NET "led_bank_0" LOC = H9;
NET "led_bank_1" LOC = E26;
NET "led_bank_2" LOC = E25;
NET "led_bank_3" LOC = F25;
NET "led_bank_4" LOC = G26;
NET "pb3_in" LOC = M22;
NET "pb4_in" LOC = N22;
NET "pb5_in" LOC = N23;
NET "pb6_in" LOC = N24;
```

Key parameters:

(1), Start pattern threshold for finding frame's time of arrival,which depends on ADC bits we used in interface module. currently the value can varies form 3 to 1500 and adjusted in variable threshold module.

(2), The chip threshold ratio for chip level synchronization, which is adjusted in variable threshold module can can change from 0.375 to 0.75 of maximum integration energy.

(3), The frame level synchronization threshold for frame level synchronization, which defines the number of tolerent errors in 60 bits, currently it can change from 8 to 20 and can be adjusted in variable threshold module.

(4), The ADC input bias shift, which defines the level the ADC quantization, currently can change from 128 to 160 and can be adjusted in variable threshold module.

(5), the ADC bits used, this value is defined in interface module such as "'data_0t ¡= RX_data[6:0];"', the syntax means we are currently using 7 bits of the ADC, and we can change to use all 8 bits by "'data_0t ¡= RX_data[7:0];"', and other 7 similar syntax should be changed accordingly.

(6), the delay cycles of the beginning of synchronization after finding the time of arrival which defined by 200ns dense pulses. The actual delay should be 2 chips time, but the delay cannot be determined because of the uncertainly of circuits' delay, currently it can change from one chips time to 3 chips time in start pattern searching module.

(7), change between 4 chips and 16 chips system, this can be done by change to accumulate 16 chips value in data process module and to increase the chip/value threshold to 4 times.some other control signal also need to adjusted in data process module.

(8), Integration window. the system is capable of increasing integration window from 10ns for time reversal system to 20 ns for pulse system, this value is adjusted in integration module.

Notes: more details can be found in file "'log.doc"'.

3.3 DAC configuration files at transmitter side

- Fujitsu_config.

Descriptions: This is the DAC configuration files folder in transmitter.

There are two configuration cases for DAC in the system 's demonstration, the first one is to use DAC memory to transmit waveform for observing focus in receiver, and the second one is to use FPGA to transmit data for data demodulation. the files are in the folder of "'data"'. Each time the registers are configured, the file "'MB86064_ab_on_for_lvds_data.txt"' need to be downloaded to the DAC and the sampling clock(500MHz) is provided by FPGA.

(1), DAC memory case. the waveform files are loaded to DAC RAM A for Q channel and RAM B for I channel. the files format should be 'vec'. there are two kind of files, the time reversal file"' "'and the pulse file"'"', actually there are many other waveforms in the folder such as ramps and sine. after RAMs loaded, the RAMs need to be synchronized, and "'MB86064_ab_on_for_lvds_data.txt"' file need to be loaded.

(2), FPGA case. the waveform is directly form FPGA output,currently we are using 8 bits of total 14 bits. we only need to load "'MB86064_ab_on_for_lvds_data.txt"' file.

Chapter 4

VNA Based CIR Measurement and Post Data Processing

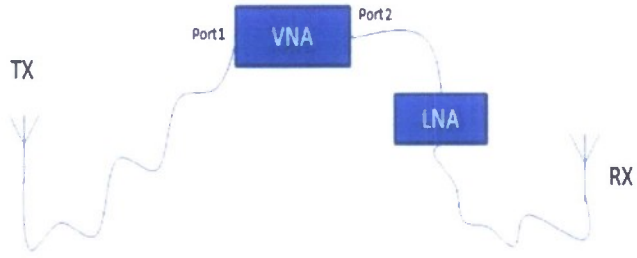

Figure 4.1: VNA measurement setup.

4.1 VNA Based CIR Measurement and Post Data Processing

A VNA can accurately measure the channel characteristics with large bandwidth. Since the Time Reversal waveform is a passband signal, it can be generated from I/Q modulating a pair of baseband waveforms with a carrier. In this project we use the VNA and post signal processing to estimate the CIR.

According to the testbed specification, the VNA frequency is set to sweep from 3.5 GHz to 4.5 GHz to cover all frequencies of interest. The frequency step is set to 1 MHz since the delay spread of the indoor channel is less than 1 μs. To get better signal-to-noise ratio (SNR) data, the number of averages is set to 256. The system setup of VNA measurement is depicted in Fig. 4.1. Two 50-feet cables with SMA connectors, a pair of antennas and a 30-dB gain low-noise-amplifier (LNA) are used. VNA transmit power levels at the two ports have to be carefully selected to prevent from damaging the LNA. The "decoupled" option has been used for flexible power setting in forward and reverse directions. Cable loss and path loss together contribute an attenuation over 100 dB, which can make the receive signal at the VNA port too weak to detect. Thus additional LNA may be necessary to compensate the losses. The connections must be tight to minimize measurement inaccuracy.

The VNA must be calibrated in prior to actual measurement. VNA calibration eliminates the impairments introduced by the connecting parts along the signal chain between Port 1 and Port 2, so that the measurement data solely reflects the characteristic of the device under test (DUT). In this measurement the DUT is the channel include the propagation channel, the LNA and the antennas. However, since the total cable loss is around 35 dB at 4 GHz, the VNA calibration is not able to compensate such amount of loss. In fact, the cable's characteristic can be measured separately and the cable's transfer function can be eliminated in off-line post signal processing.

The total filtering effect of the system's RF front-ends at both sides should be taken into account. It has been found that this equivalent filter in baseband representation is dominated by the D/A's interpolating filer which is measurable. The measured data of the interpolating filter has been combined with the VNA measurement data to make the CIR estimate more accurate.

After measurement, we load the frequency domain data into a Matlab program and get the baseband time reversal I/Q waveform. The VNA data is denoted as $C[f], f_l \leq f \leq f_h$, with center frequency f_0. We shift the signal to

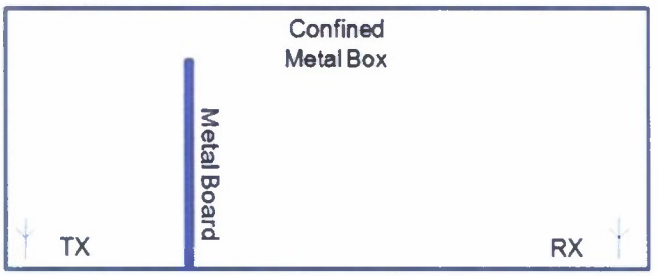

Figure 4.2: Environment setup for DAC test.

baseband and get $C[f - f_0]$. Then, we can get the baseband I/Q waveform according to the following equations:

$$
\begin{aligned}
&c[t] \underline{\underline{IFTC}} [f - f_0] \\
&c[t] = a[t] + jb[t] \\
&s[t] = \mathrm{Re}\left\{c[t] e^{j2\pi f_0 t}\right\} = \mathrm{Re}\left\{(a[t] + jb[t])(\cos[2\pi f_0 t] + j\sin[2\pi f_0 t])\right\} = a[t]\cos[2\pi f_0 t] - b[t]\sin[2\pi f_0 t]
\end{aligned}
$$

where $s[t]$ is the modulated signal, whose positive frequency is the conjugation of the channel transfer function. So, we have baseband I/Q waveform as:

$$
\begin{aligned}
I[t] &= a[t] \\
Q[t] &= -b[t]
\end{aligned}
$$

This baseband I/Q waveform is loaded into D/A's memory to test whether a desired signal can be generated. An experiment in a metallic box was conducted for this verification purpose. The layout of the experiment is illustrated in Fig. 4.2. A transmit antenna and a receive antenna are blocked by a metal board. The whole box serves as a confined propagation environment. The magnitude of a CIR measured in the box is depicted in Fig. 4.1, and the quantized I/Q waveform is shown in Fig. 4.4 as well as Fig. 4.5. The received waveform at the detector is observed with an oscilloscope. According to time reversal phenomenon, one strong envelope peak should be observed. Fig. 4.6 shows an incorrect result. This is due to the unmatched I/Q waveform caused by a 50 MHz LO shift. Fig. 4.7 shows the result with matched transmit waveform. At this point we can say that the acquired I/Q waveforms and the system work properly.

Other issues to consider include the waveform length truncation and vertical clipping due to the implementation limits in the digital FIR filter. For time reversal precoding, although the actual length of CIR may excess 160 ns, the testbed supports waveforms of length 160 ns (160 points) at 1-Gsps sampling rate. To maximize the performance under this length limit, a 160-ns long sliced waveform with maximum energy is selected. It is reasonable to set a 160-ns length limit because the delay spread in a typical indoor environment is less than 160 ns. The waveforms can be overlapped as the chip duration is 40 ns, less than 160 ns. Vertical clipping may occur since the number of quantization bits is limited. Chapter 2.4 has provided a phase rotation method to maximally utilize the quantization ability and avoid clipping. As will be discussed in the system trail section, minor clipping is tolerable in time reversal precoding. In other words, we can make phase rotation solely based on a single chip waveform, without considering the impact of chip waveform overlapping. This simplifies the process but usually leads to some insignificant clipping.

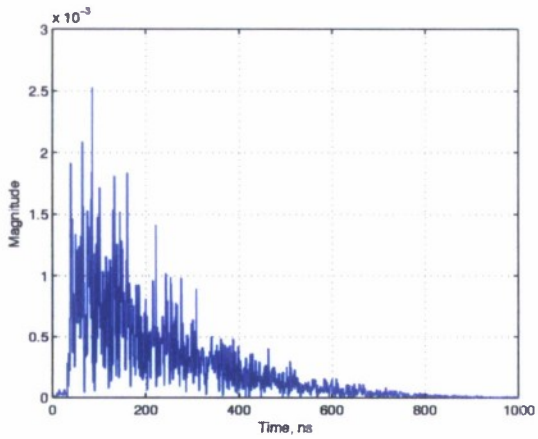

Figure 4.3: Envelope of the channel impulse response in the metallic box.

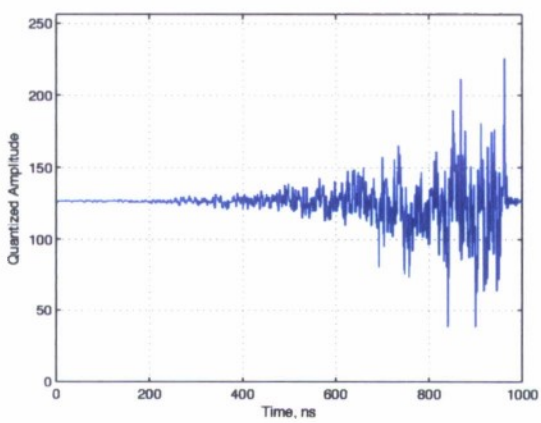

Figure 4.4: Quantized I channel.

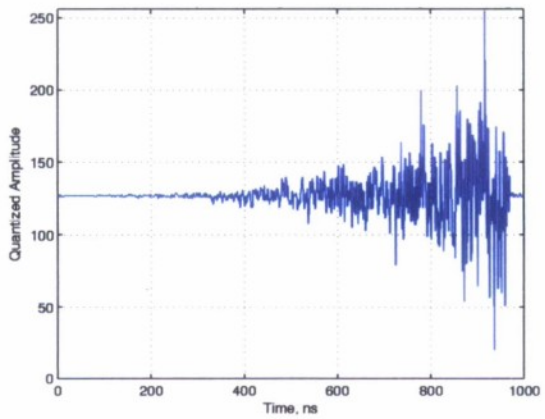

Figure 4.5: Quantized Q channel.

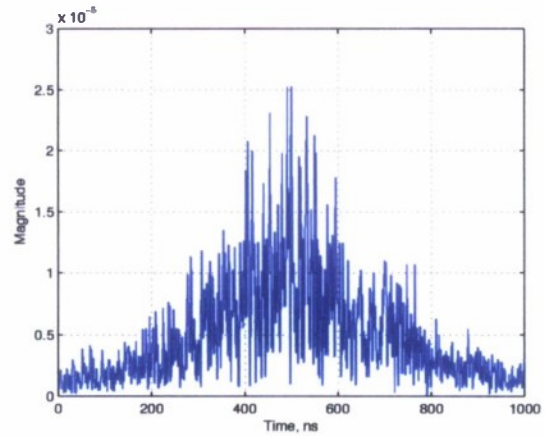

Figure 4.6: Received waveform with 50 MHz frequency offset of the transmit waveform caused by LO.

Figure 4.7: Received waveform with matched transmit waveform. Measured with DSO at the detector.

4.1.1 Main Files for Channel Measurement and Post Data Processing

All files are placed under the same folder.

Input files:

- CHANNEL.cti

Description: This file stores the S21 parameter of the channel. It is the data file obtained from VNA measurement. 'CHANNEL' is user defined.

- cable_3_5.cti

Description: This is the S21 parameter of two 50 ft SMA cables connecting together.

- DAC_1ns.mat

Description: This is the time domain 1 ns pulse generated by DAC. It is captured by DPO.

Main processing file:

- Load_FIR_Coef_VNA.m

Description: This is the main file for post data processing. Input files are loaded and several output files for FPGA and DAC will be generated.

Output files:

- Xpt_1_I_FPGA.txt, Xpt_1_Q_FPGA.txt,

Description: These are the I/Q waveform template files loaded in FPGA. X is the number of points for the truncated CIR.

- FILENAME_I.vec, FILENAME_Q.vec

Description: These are the I/Q waveform template files loaded in DAC. FILENAME is the filename of the data file storing S21 parameters.

Part II

Theoretical Work

Chapter 5

Wideband Waveform Design using Energy Detector Receiver with Practical Considerations

5.1 Introduction

Motivated by increasing demand for cheap wireless sensor networks, a wideband radio system combining waveform precoding and simple receivers is considered in this paper. Among receiver options is energy detector receiver that is attractive because of its compromising between performance and simplicity. The central idea is to pair the simple receiver with a sophisticated transmitter which is able to transmit an optimal waveform to achieve performance gain. Such type of transmitter featured by an arbitrary waveform generator has been tested in our laboratorial UWB test-bed, thanks to the recent technology advanced in semiconductors. Although multiple of Nyquist sampling rate has been feasible for Giga-Hertz bandwidth, reduction in quantization and transmitted peak power is very desired from implementation perspective.

Four practical cases are considered in this paper, which are all critical issues when we built our laboratorial UWB test-bed. There should be a tradeoff between energies within and outside integration window if inter-symbol-interference (ISI) has to be concerned. Because of the hardware limitation or implementation simplicity, a low-resolution quantization has to be employed sometimes. Thus the design for binary waveform and ternary waveform are studied. Meanwhile, binary waveform or ternary waveform can be treated as one kind of robust waveform, if channel estimation is not accurate. For waveform design in the digital domain, one of the challenging issues is Peak-to-Average Power Ratio (PAPR) because of the nonlinear devices used in the system. The transmitted peak power should be reduced such that the nonlinear devices can work in the proper region. All the practical cases are formulated as the corresponding optimization problems. By using the advanced optimization tool, the optimal solution or suboptimal solution to these optimization problems can be achieved. The work in this paper can not only make the contribution to the theoretical research in waveform optimization and synthesis but also give the guidance for the real system design and implementation.

The rest of the paper is organized as follows. The system is described in Section II. Practical considerations for wideband waveform design are presented in Section III. Numerical results are provided in Section IV, followed by some remarks given in Section V.

5.2 System Description and Optimal Waveform

The system architecture is shown in Figure 5.1. We limit our discussion to a single-user scenario, and consider the transmitted signal with OOK modulation given by

$$s(t) = \sum_{j=-\infty}^{\infty} d_j p(t - jT_b) \tag{5.1}$$

where T_b is the symbol duration, $p(t)$ is the transmitted symbol waveform defined over $[0, T_p]$ and $d_j \in \{0, 1\}$ is j-th transmitted bit. Without loss of generality, assume the minimal propagation delay is equal to zero. The energy of $p(t)$ is E_p,

$$\int_0^{T_p} p^2(t)\, dt = E_p \tag{5.2}$$

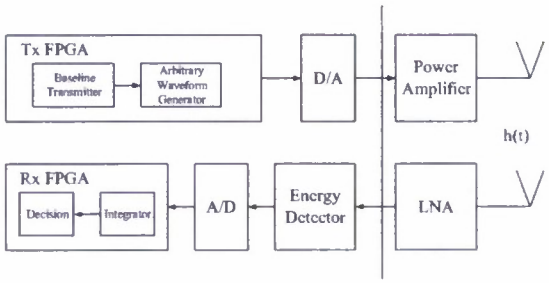

Figure 5.1: System architecture.

The received noise-polluted signal at the output of the receiver front-end filter is

$$\begin{aligned} r(t) &= h(t) \otimes s(t) + n(t) \\ &= \sum_{j=-\infty}^{\infty} d_j x(t - jT_b) + n(t), \end{aligned} \tag{5.3}$$

where $h(t), t \in [0, T_h]$ is the multipath impulse response that takes into account the effect of channel impulse response, the RF front-ends in the transceivers including antennas. $h(t)$ is available at the transmitter. "$\otimes$" denotes convolution operation. $n(t)$ is a low-pass additive zero-mean Gaussian noise with one-sided bandwidth W and one-sided power spectral density N_0. $x(t)$ is the received noiseless symbol-"1" waveform defined as

$$x(t) = h(t) \otimes p(t) \tag{5.4}$$

We further assume that $T_b \geq T_h + T_p \stackrel{\text{def}}{=} T_x$, i.e. no existence of ISI.

An energy detector performs square operation to $r(t)$. Then the integrator does the integration over a given integration window T_I. Corresponding to the time index k, the k-th decision statistic at the output of the integrator is given by

$$z_k = \int_{kT_b+T_{I0}}^{kT_b+T_{I0}+T_I} r^2(t)dt \tag{5.5}$$

$$= \int_{kT_b+T_{I0}}^{kT_b+T_{I0}+T_I} (d_k x(t - kT_b) + n(t))^2 dt \tag{5.6}$$

where T_{I0} is the starting time of integration for each symbol and $0 \leq T_{I0} < T_{I0} + T_I \leq T_x \leq T_b$.

An approximately equivalent SNR for the energy detector receiver, which provides the same detection performance when applied to a coherent receiver, is given as

$$\text{SNR}_{\text{eq}} = \frac{2\left(\int_{T_{I0}}^{T_{I0}+T_I} x^2(t)\, dt\right)^2}{2.3 T_I W N_0^2 + N_0 \int_{T_{I0}}^{T_{I0}+T_I} x^2(t)\, dt} \tag{5.7}$$

For best performance, the equivalent SNR SNR_{eq} should be maximized. Define,

$$E_I = \int_{T_{I0}}^{T_{I0}+T_I} x^2(t)\, dt \tag{5.8}$$

For given T_I, N_0 and W, SNR_{eq} is the increasing function of E_I. So the maximization of SNR_{eq} in Equation 5.7 is equvalent to the maximization of E_I in Equation 5.8.

So the optimization problem to get the optimal $\mathbf{p}$ is shown below,

$$\begin{array}{l} \max \int_{T_{I0}}^{T_{I0}+T_I} x^2(t)\,dt \\ s.t. \int_0^{T_p} p^2(t)\,dt = E_p \end{array} \tag{5.9}$$

In order to solve the optimization problem 5.9, numerical approach is employed in this paper. In other words, $p(t)$, $h(t)$ and $x(t)$ are uniformly sampled (assumed at Nyquist rate), and the optimization problem 5.9 will be converted to its corresponding discrete-time form. Assume the sampling period is T_s. $T_p/T_s = N_p$, $T_h/T_s = N_h$ and $T_x/T_s = N_x$. So $N_x = N_p + N_h$.

$p(t)$, $h(t)$ and $x(t)$ are represented by $p_i, i = 0, 1, \ldots, N_p$, $h_i, i = 0, 1, \ldots, N_h$ and $x_i, i = 0, 1, \ldots, N_x$ respectively.

Define,

$$\mathbf{p} = [p_0\ p_1\ \cdots\ p_{N_p}]^T \tag{5.10}$$

and

$$\mathbf{x} = [x_0\ x_1\ \cdots\ x_{N_x}]^T \tag{5.11}$$

Construct channel matrix $\mathbf{H}_{(N_x+1)\times(N_p+1)}$,

$$(\mathbf{H})_{i,j} = \begin{cases} h_{i-j}, 0 \le i-j \le N_h \\ 0, \quad else \end{cases} \tag{5.12}$$

where $(\bullet)_{i,j}$ denotes the entry in the i-th row and j-th column of the matrix or vector. Meanwhile, for vector, taking $\mathbf{p}$ as an example, $(\mathbf{p})_{i,1}$ is equivalent to p_{i-1}.

The matrix expression of Equation 5.4 is,

$$\mathbf{x} = \mathbf{H}\mathbf{p} \tag{5.13}$$

and the constraint in the optimization problem 5.9 can be expressed as,

$$\|\mathbf{p}\|_2^2 T_s = E_p \tag{5.14}$$

where "$\|\bullet\|_2$" denotes the norm-2 of the vector. In order to make the whole paper consistent, we further assume,

$$\|\mathbf{p}\|_2^2 = 1 \tag{5.15}$$

Let $T_I/T_s = N_I$ and $T_{I0}/T_s = N_{I0}$. The entries in $\mathbf{x}$ within integration window constitute $\mathbf{x}_I$ as,

$$\mathbf{x}_I = [x_{N_{I0}}\ x_{N_{I0}+1}\ \cdots\ x_{N_{I0}+N_I}]^T \tag{5.16}$$

and E_I in Equation 5.8 can be equivalently shown as,

$$E_I = \|\mathbf{x}_I\|_2^2 T_s \tag{5.17}$$

Simply dropping T_s in E_I will not affect the optimization objective, so E_I is redefined as,

$$E_I = \|\mathbf{x}_I\|_2^2 \tag{5.18}$$

Similar to Equation 5.13, $\mathbf{x}_I$ can be obtained by,

$$\mathbf{x}_I = \mathbf{H}_I \mathbf{p} \tag{5.19}$$

where $(\mathbf{H}_I)_{i,j} = (\mathbf{H})_{N_{I0}+i,j}$ and $i = 1, 2, \ldots, N_I + 1$ as well as $j = 1, 2, \ldots, N_p + 1$.

The optimization problem 5.9 can be represented by its discrete-time form as,

$$\begin{aligned} &\max E_I \\ &s.t. \ \|\mathbf{p}\|_2^2 = 1 \end{aligned} \tag{5.20}$$

The optimal solution $\mathbf{p}^*$ for the optimization problem 5.20 is the dominant eigen-vector in the following eigen-function,

$$\mathbf{H}_I^T \mathbf{H}_I \mathbf{p} = \lambda \mathbf{p} \tag{5.21}$$

Furthermore, E_I^* will be obtained by Equation 5.18 and Equation 5.19.

5.3 Waveform Design with Practical Considerations

5.3.1 Tradeoff between Energies Within And Outside Integration Window

The energy outside the integration window needs to be concerned sometimes, say, when ISI has to be considered. In order to reduce ISI, the energies within and outside integration window should be balanced, which means the energy within integration window should be maximized and the energy outside integration window should be minimized.

The entries in $\mathbf{x}$ outside integration window constitute $\mathbf{x}_{\bar{I}}$ as,

$$\mathbf{x}_{\bar{I}} = [x_0 \ \cdots \ x_{N_{I0}-1} \ x_{N_{I0}+N_I+1} \ \cdots \ x_{N_x}]^T \tag{5.22}$$

and the energy outside integration window $E_{\bar{I}}$ can be expressed as,

$$E_{\bar{I}} = \|\mathbf{x}_{\bar{I}}\|_2^2 \tag{5.23}$$

Similar to Equation 5.19, $\mathbf{x}_{\bar{I}}$ can be obtained by,

$$\mathbf{x}_{\bar{I}} = \mathbf{H}_{\bar{I}} \mathbf{p} \tag{5.24}$$

where $(\mathbf{H}_{\bar{I}})_{i,j} = (\mathbf{H})_{i,j}$ when $i = 1, \ldots, N_{I0}$ and $(\mathbf{H}_{\bar{I}})_{i-(N_I+1),j} = (\mathbf{H})_{i,j}$ when $i = N_{I0} + N_I + 2, \ldots, N_x + 1$ as well as $j = 1, 2, \ldots, N_p + 1$.

In order to balance energies within and outside integration window, the tradeoff factor α is introduced. The range of α is from 0 to 1. Given α, the optimization problems is formulated as follows,

$$\begin{aligned} &\max \alpha E_I - (1-\alpha) E_{\bar{I}} \\ &s.t. \ \|\mathbf{p}\|_2^2 = 1 \end{aligned} \tag{5.25}$$

The optimal solution $\mathbf{p}^*$ for the optimization problem 5.25 is the dominant eigen-vector in the following eigen-function,

$$\left[\alpha \mathbf{H}_I^T \mathbf{H}_I - (1-\alpha) \mathbf{H}_{\bar{I}}^T \mathbf{H}_{\bar{I}}\right] \mathbf{p} = \lambda \mathbf{p} \tag{5.26}$$

5.3.2 Binary Waveform

If the transmitted waveform is constrained to the binary waveform because of the hardware limitation or implementation simplicity, which means $p_i, i = 0, 1, \ldots, N_p$ is equal to $-\frac{1}{\sqrt{1+N_p}}$ or $\frac{1}{\sqrt{1+N_p}}$, then the optimization problem is expressed as,

$$\begin{aligned} &\max E_I \\ &s.t.[(\mathbf{p})_{i,1}]^2 = \tfrac{1}{1+N_p}, i = 0, 1, \ldots, N_p \end{aligned} \tag{5.27}$$

One suboptimal solution $\mathbf{p}_{b1}^*$ to the optimization problem 5.27 is derived from the optimal solution $\mathbf{p}^*$ of the optimization problem 5.20. When $\mathbf{p}^*$ is obtained, then

$$(\mathbf{p}_{b1}^*)_{i,1} = \begin{cases} \frac{1}{\sqrt{1+N_p}}, (\mathbf{p}^*)_{i,1} \geq 0 \\ -\frac{1}{\sqrt{1+N_p}}, (\mathbf{p}^*)_{i,1} < 0 \end{cases} \tag{5.28}$$

This simple method can lead to the optimal solution to the optimization problem 5.27 when $T_I \to 0$, which can be proofed by CauchySchwarz inequality, but if T_I is greater than zero, there is still a improvement potential to this suboptimal solution obtained from Equation 5.28.

It is well known that the optimization problem 5.27 is Quadratically Constrained Quadratic Program (QCQP) and general QCQP is NP-hard, so a semidefinite relaxation method is proposed to give the suboptimal solution to this optimization problem.

Define,

$$\mathbf{P} = \mathbf{p}\mathbf{p}^T \tag{5.29}$$

P should be a symmetric positive semidefinite matrix, i.e. $\mathbf{P} \succ= 0$ and rank of **P** should be equal to 1. Reformulate E_I as,

$$\begin{aligned} E_I &= \mathbf{p}^T \mathbf{H}_I^T \mathbf{H}_I \mathbf{p} && (5.30) \\ &= \text{trace}\left(\mathbf{H}_I^T \mathbf{H}_I \mathbf{p}\mathbf{p}^T\right) && (5.31) \\ &= \text{trace}\left(\mathbf{H}_I^T \mathbf{H}_I \mathbf{P}\right) && (5.32) \end{aligned}$$

Rank constraint is nonconvex constraint, so after dropping it, QCQP is relaxed to the Semidefinite Program (SDP),

$$\begin{aligned} &\max \text{trace}\left(\mathbf{H}_I^T \mathbf{H}_I \mathbf{P}\right) \\ &s.t. (\mathbf{P})_{i,i} = \tfrac{1}{1+N_p}, i = 0, 1, \ldots, N_p \\ &\quad \mathbf{P} \succ= 0 \end{aligned} \tag{5.33}$$

The optimal solution $\mathbf{P}^*$ of the optimization problem 5.33 can be obtained by using CVX tool from Stanford University and the value of the objective function in the optimizatiion problem 5.33 gives the upper bound of the optimal value in the optimization problem 5.27. Project the dominant eigen-vector of $\mathbf{P}^*$ on $-\frac{1}{\sqrt{1+N_p}}$ and $\frac{1}{\sqrt{1+N_p}}$ based on Equation 5.28, the suboptimal solution $\mathbf{p}_{b2}^*$ is achieved using the method proposed in Semidefinite Optimization with Applications in Sparse Multivariate Statistics.

Finally, the designed binary waveform is,

$$\mathbf{p}_b^* = \arg \max_{\mathbf{p} \in \{\mathbf{p}_{b1}^*, \mathbf{p}_{b2}^*\}} \mathbf{p}^T \mathbf{H}_I^T \mathbf{H}_I \mathbf{p} \tag{5.34}$$

5.3.3 Ternary Waveform

If the transmitted waveform is constrained to the ternary waveform, which means $p_i, i = 0, 1, \ldots, N_p$ is equal to three levels, i.e. $-c$, 0 or c, then the optimization problem is expressed as,

$$\begin{aligned} &\max E_I \\ &s.t.[(\mathbf{p})_{i,1}]^2 = c^2 \text{ or } 0, i = 0, 1, \ldots, N_p \\ &\quad \|\mathbf{p}\|_2^2 = 1 \end{aligned} \tag{5.35}$$

where the value of c will be determined later.

The optimization problem 5.35 is still NP-hard and can be approximately reformulated as,

$$\begin{aligned} &\max E_I \\ &s.t. \text{Cardinality}\,(\mathbf{p}) \leq k \\ &\quad 1 \leq k \leq N_p + 1 \quad \|\mathbf{p}\|_2^2 = 1 \end{aligned} \tag{5.36}$$

where Cardinality $(\mathbf{p})$ denotes the number of non-zero entries of $\mathbf{p}$ and cardinality constraint is also a nonconvex constraint.

Because k is the integer number between 1 and $N_p + 1$, the optimization problem 5.36 can be decomposed into $N_p + 1$ independent and parallel sub-problems and each sub-problem is shown as,

$$\begin{aligned} &\max E_I \\ &s.t. \text{Cardinality}\,(\mathbf{p}) \leq k \\ &\quad \|\mathbf{p}\|_2^2 = 1 \end{aligned} \tag{5.37}$$

where k is equal to $1, 2, \cdots,$ or $N_p + 1$;

The sub-problems 5.37 can be solved in parallel and then the solutions are combined to get the solution of the original optimization problem 5.35. Reuse the definition in Equation 5.29 and the sub-problem 5.37 can converted to the following SDP by semidefinite relaxation combined with l1 heuristic proposed in Semidefinite Optimization with Applications in Sparse Multivariate Statistics.

$$\begin{aligned} &\max \text{trace}\left(\mathbf{H}_I^T \mathbf{H}_I \mathbf{P}\right) \\ &s.t. \text{trace}\,(\mathbf{P}) = 1 \\ &\quad \mathbf{a}^T |\mathbf{P}|\, \mathbf{a} \leq k \\ &\quad \mathbf{P} \succ= 0 \end{aligned} \tag{5.38}$$

where $\mathbf{a}$ is all-1 column vector and,

$$\|\mathbf{p}\|_2^2 = \mathbf{p}^T \mathbf{p} \tag{5.39}$$

$$= \text{trace}\left(\mathbf{p}\mathbf{p}^T\right) \tag{5.40}$$

$$= \text{trace}\,(\mathbf{P}) \tag{5.41}$$

The CVX tool from Stanford University is also operated to get the optimal solution $\mathbf{P}_k^*$ of SDP 5.38. From the dominant eigen-vector $\mathbf{p}_k^*$ of $\mathbf{P}_k^*$ and the threshold p_{thk}, the solution for the sub-problem 5.37 can be achieved as,

$$(\mathbf{p}_{\text{tk}}^*)_{i,1} = \begin{cases} c_k, (\mathbf{p}_k^*)_{i,1} > p_{\text{thk}} \\ 0, \left|(\mathbf{p}_k^*)_{i,1}\right| \leq p_{\text{thk}} \\ -c_k, (\mathbf{p}_k^*)_{i,1} < p_{\text{thk}} \end{cases} \tag{5.42}$$

where

$$\begin{aligned} p_{\text{thk}} &= \arg\max_{\{p_{\text{th}}\}} \left(\mathbf{p}_{\text{tk}}^{*}\right)^{T} \mathbf{H}_{I}^{T} \mathbf{H}_{I} \mathbf{p}_{\text{tk}}^{*} \\ & s.t. \text{Cardinality}\left(\mathbf{p}_{\text{tk}}^{*}\right) \leq k \end{aligned} \tag{5.43}$$

and

$$c_k = \frac{1}{\sqrt{\text{Cardinality}\left(\mathbf{p}_{\text{tk}}^{*}\right)}} \tag{5.44}$$

Finally, the designed ternary waveform is,

$$\mathbf{p}_{t}^{*} = \arg \max_{\mathbf{p} \in \left\{\mathbf{p}_{\text{tk}}^{*}, k=1,2,\ldots,N_p+1\right\}} \mathbf{p}^{T} \mathbf{H}_{I}^{T} \mathbf{H}_{I} \mathbf{p} \tag{5.45}$$

5.3.4 Peak-to-Average Power Ratio

PAPR is one of major concerns in waveform design. Because of nonlinearity caused by nonlinear devices such as Digital-to-Analog Converter (DAC) and Power Amplifier (PA), maximal transmitted power has to be backed up, resulting in inefficient utilization. PAPR in OFDM has been well studied. In this paper, PAPR is handled under a unified optimization framework. It is defined as,

$$\text{PAPR} = \frac{\|\mathbf{p}\|_{\infty}^{2}}{\|\mathbf{p}\|_{2}^{2} \Big/ (N_p + 1)} \tag{5.46}$$

where

$$\|\mathbf{p}\|_{\infty} = \max\left(|p_0|, |p_1|, \cdots, \left|p_{N_p}\right|\right) \tag{5.47}$$

If $\|\mathbf{p}\|_{2}^{2}$ is given, reducing PAPR is equivalent to setting the upper bound for $\|\mathbf{p}\|_{\infty}$. So the optimization problem can be expressed as,

$$\begin{aligned} &\max E_I \\ &s.t. \|\mathbf{p}\|_{2}^{2} = 1 \\ &\quad \|\mathbf{p}\|_{\infty} \leq \text{ub} \end{aligned} \tag{5.48}$$

The bound constraint $\|\mathbf{p}\|_{\infty} \leq \text{ub}$ can also be written as,

$$-\text{ub} \leq p_i \leq \text{ub}, i = 0, 1, \ldots, N_p \tag{5.49}$$

which can be further simplified as,

$$p_i^2 \leq (\text{ub})^2, i = 0, 1, \ldots, N_p \tag{5.50}$$

Reuse the definition in Equation 5.29, the optimization problem 5.46 can be relaxed to SDP,

$$\begin{aligned} &\max \text{trace}\left(\mathbf{H}_{I}^{T} \mathbf{H}_{I} \mathbf{P}\right) \\ &s.t. \left(\mathbf{P}\right)_{i,i} \leq (\text{ub})^2, i = 0, 1, \ldots, N_p \\ &\quad \text{trace}\left(\mathbf{P}\right) = 1 \\ &\quad \mathbf{P} \succ = 0 \end{aligned} \tag{5.51}$$

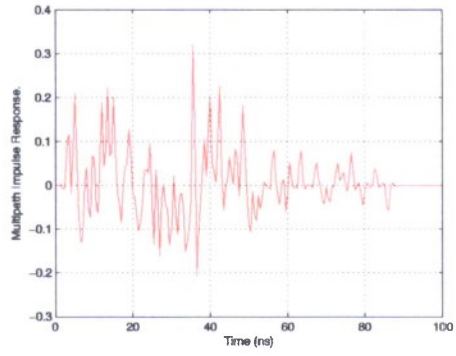

Figure 5.2: Multhpath impulse response.

By CVX tool from Stanford University, the optimal solution $\mathbf{P}^*$ of the optimization problem 5.51 is obtained. If the rank of $\mathbf{P}^*$ is equal to 1, then the dominant eigen-vector of $\mathbf{P}^*$ will be the optimal solution $\mathbf{p}^*$ for the optimization problem 5.48. But if the rank of $\mathbf{P}^*$ is not equal to 1, the dominant eigen-vector of $\mathbf{P}^*$ can not be treated as the optimal solution for the optimization problem 5.48, because of the violation of bound constraint.

So a computationally-efficient iterative algorithm is proposed to get the suboptimal solution $\mathbf{p}^*$ to the optimization problem 5.48 as follows.

1. Initialization: $P = 1$, $\mathbf{H}_0 = \mathbf{H}_I^T \mathbf{H}_I$ and $\mathbf{p}^*$ is set to be all-0 column vector.

2. Solve the following optimization problem to get the optimal $\mathbf{q}$.

$$\begin{aligned} &\max \mathbf{q}^T \mathbf{H}_0 \mathbf{q} \\ &s.t. \ \|\mathbf{q}\|_2^2 = P \end{aligned} \tag{5.52}$$

3. Find i, such that $|q_i|$ is the maximal value in the set $\{|q_j| \, | |q_j| > \text{ub}\}$. If $\{i\} = \emptyset$, then the algorithm is terminated and $\mathbf{p}^* := \mathbf{p}^* + \mathbf{q}$. Otherwise go to step 4.

4. If q_i is greater than zero, then $(\mathbf{p}^*)_{i,1}$ is set to be ub. Otherwise $(\mathbf{p}^*)_{i,1}$ is set to be $-\text{ub}$.

5. $P := P - (\text{ub})^2$ and set $(\mathbf{H}_0)_{i,j}, j = 1, 2, \ldots, N_p + 1$ and $(\mathbf{H}_0)_{j,i}, j = 1, 2, \ldots, N_p + 1$ all to zeros. Go to step 2.

5.4 Numerical Results

Figure 5.2 shows the multipath impulse response $h(t)$ under investigation in this paper and the energy of $h(t)$ is normalized. $T_s = 0.5ns$, $T_h = 100ns$, $T_p = 100ns$ and $T_{I0} + \frac{T_I}{2} = 100ns$.

Figure 5.3 shows the energy E_I in the integration window defined by Equation 5.18 when different tradeoff factor $\alpha's$ are chosen. Figure 5.4 shows the the corresponding ratio of energies $E_I/E_{\bar{I}}$ within and outside integration window. When α is close to 1, the energy within integration window will have more weight. The optimization objective is

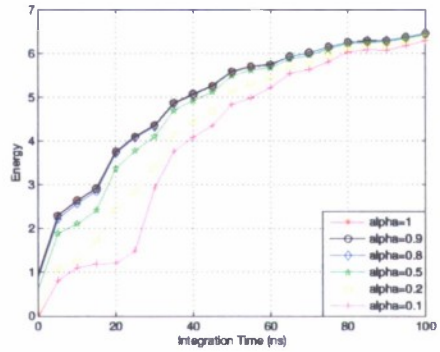

Figure 5.3: The energy within the integration window.

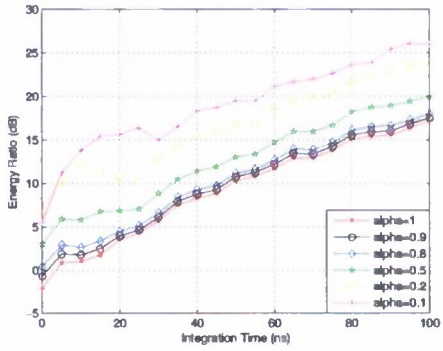

Figure 5.4: The ratio of energies within and outside integration window.

to maximize the energy within integration window. Thus the larger α, the more energy within integration window. However, when α is close to 0, the optimization objective will be to minimize the energy outside integration window. From Figure 5.4, the trend is, the smaller α, the bigger the ratio of energies within and outside integration window. This results give us the hint to design the system with proper parameters if ISI is introduced.

For PAPR, if $T_I \rightarrow 0$ and ub is set to be 0.4, the designed waveforms are shown in Figure 5.5. In this case, the optimal waveform without consideration of PAPR, the waveform obtained by SDP and the waveform achieved by iterative algorithm are the same. The maximal eigen-value of $\mathbf{P}^*$ in the optimization problem 5.51 approaches 1, which means the rank of $\mathbf{P}^*$ is 1. So waveform obtained by SDP will not violate the bound constraint. Meanwhile, if $T_I \rightarrow 0$ and ub is not tight, the optimal waveform is the time reversed multipath impulse response. if $T_I \rightarrow 0$ and ub is set to be 0.15, the designed waveforms are shown in Figure 5.6. The optimal waveform without consideration of PAPR is still the time reversed multipath impulse response. The waveform obtained by SDP is equivalent to that achieved by iterative algorithm, which means both of two methods give the optimal solution under PAPR constraint.

If T_I is equal to $2ns$ and ub is set to be 0.15, the designed waveforms are shown in Figure 5.7. Because the rank of $\mathbf{P}^*$ in the optimization problem 5.51 is equal to 1, the waveform obtained by SDP gives the optimal solution. Meanwhile the waveform achieved by iterative algorithm gives the suboptimal solution and the ratio of energies within the integration window by the suboptimal waveform and the optimal waveform with PAPR is 0.99. If ub is set to be 0.1, the designed waveforms are shown in Figure 5.8. The rank of $\mathbf{P}^*$ in the optimization problem 5.51 is

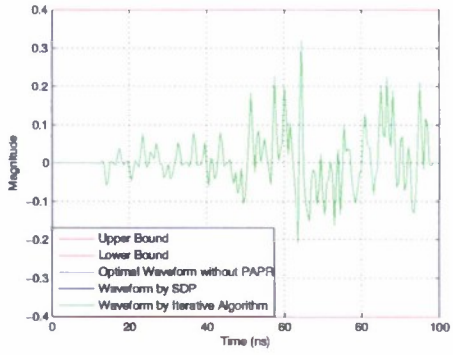

Figure 5.5: The designed waveforms if $T_I \to 0$ and ub is set to be 0.4.

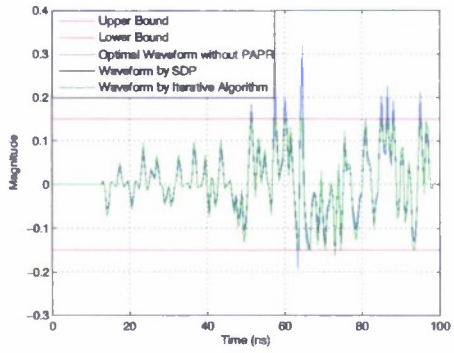

Figure 5.6: The designed waveforms if $T_I \to 0$ and ub is set to be 0.15.

equal to 2 in this case, so the waveform obtained by SDP does not give the solution. It is easy to see from Figure 5.8 that this waveform violates the bound constraints in some samples.

In order to compare the performance of optimal waveform, binary waveform and ternary waveform, the down-sampled multipath impulse response in Figure 5.2 is used. Figure 5.9 shows the energies when optimal waveform, binary waveform and ternary waveform are employed. Because of SDP relaxation, the proposed algorithms to get binary waveform or ternary waveform can not guarantee optimum from energy's point of view all the times. However, energy of the suboptimal binary or ternary waveform approaches that of the optimal binary or ternary waveform very well. Take this case as an example, the suboptimal ternary waveform catches at least 94 percent energy compared the optimal ternary waveform, while for binary waveform, this number is well above 0.99. Meanwhile, ternary waveform can obtain more than 80 percent energy from the optimal waveform, which makes ternary waveform competent in this kind of system with energy detector receiver.

5.5 Conclusion

Wideband waveform optimization with energy detector receiver has been studied in this paper. This work is a part of our effort in searching for simple-receiver solutions with enhanced performance. The contribution of this paper

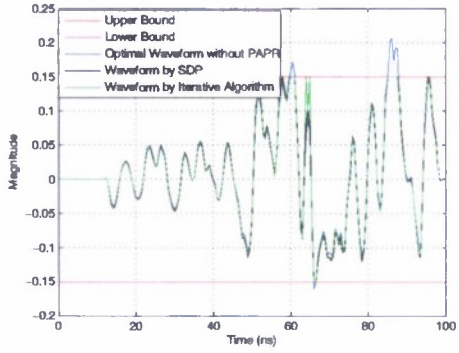

Figure 5.7: The designed waveforms if T_I is equal to $2ns$ and ub is set to be 0.15.

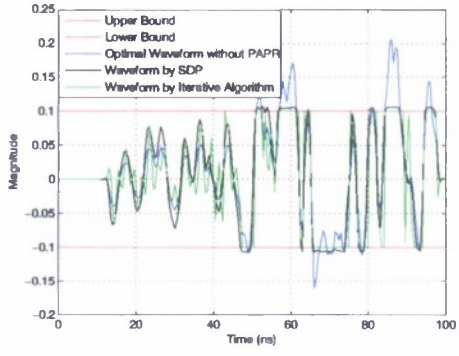

Figure 5.8: The designed waveforms if T_I is equal to $2ns$ and ub is set to be 0.1.

is to combine waveform design and optimization with the practical situations. Four practical cases are considered in this paper, i.e. the tradeoff between energies within and outside integration window, binary waveform, ternary waveform and PAPR. The methods and results of this paper can give us the guidance for the real system design and implementation.

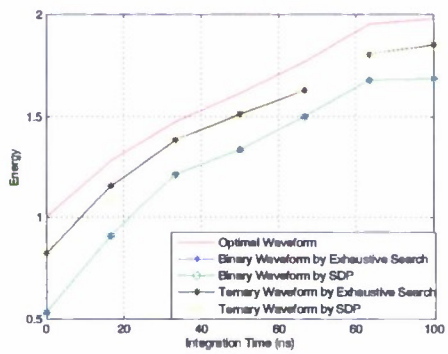

Figure 5.9: The energies when optimal waveform, binary waveform and ternary waveform are employed.